

Run PySpark Word Count example on Google Cloud Platform using Dataproc

Overview

This word count example is similar to the one introduced earlier. It will use the Shakespeare dataset in BigQuery. The only difference is that instead of using Hadoop, it uses PySpark which is a Python library for Spark.

Step 1: create the output table in BigQuery

We need a table to store the output of our Map Reduce procedure.

- Select your project or create a new one, remember to enable billing
- Go to Big Query
- Create a dataset, then a table using the following schema

The screenshot shows the BigQuery 'Table Details' page for a table named 'wc1'. On the left, the navigation pane shows the project 'cuong-project1' and the dataset 'wc_class'. The main content area has tabs for 'Schema', 'Details', and 'Preview'. The 'Schema' tab is active, displaying a table with two columns: 'word' (STRING, NULLABLE) and 'word_count' (INTEGER, NULLABLE). Below the table is an 'Add New Fields' button.

Schema	Details	Preview	
word	STRING	NULLABLE	Describe this field...
word_count	INTEGER	NULLABLE	Describe this field...

Example: my project is “cuong-project1”, dataset is “wc_class”, and output table is “wc1”. The output table has two fields: word and word_count. We will need these fields to store the output.

Step 2: create a cluster in Dataproc and Google Cloud Storage

Go to Dataproc and create a cluster similar to our previous tutorial. It should look like this

The screenshot shows the Dataproc 'Clusters' page. At the top is a search bar with the text 'Search clusters, press Ent'. Below is a table with columns: Name, Zone, Total worker nodes, Cloud Storage staging bucket, Created, and Status. One cluster is listed: 'cluster-1' in zone 'us-west1-a' with 3 worker nodes. The 'Cloud Storage staging bucket' column contains a long URL. The cluster was created on May 9, 2017, at 9:57:29 AM and is in a 'Running' state.

Name	Zone	Total worker nodes	Cloud Storage staging bucket	Created	Status
cluster-1	us-west1-a	3	dataproc-e4225d08-23a8-481f-aff9-5205024119a5-us	May 9, 2017, 9:57:29 AM	Running

Now click on the link in the “Cloud Storage staging bucket” column, it will bring you to your Cloud Storage. This is where we will upload our python code, which will then give us a link to submit a job to the cluster.

Step 3: modify the code and upload it to Cloud Storage

You can download the python code 'sparkwc.py' from our piazza website.

You can look at the code here <https://cloud.google.com/dataproc/examples/bigquery-connector-spark-example> (remember to click on the PySpark tab instead of Scala)



```
input_directory =  
'gs://{}/hadoop/tmp/bigquery/pyspark_input'.format(bucket)
```

Replace the {} symbols with your Cloud Storage Bucket id.

```
#output_dataset = 'wordcount_dataset'  
#output_table = 'wordcount_table'
```

Set the corresponding dataset name and table name of your output table (created in Step 1).

```
output_directory =
```

similar to input_directory, assign the correct Cloud Storage Bucket id here.

Then, upload the python file to your Cloud Storage and get the link to it. We will use this link as input to our PySpark job.

Browser			
Buckets / dataproc-e4225d08-23a8-481f-aff9-5205024119a5-us			
☐	Name	Size	Type
☐	google-cloud-dataproc-metainfo/	—	Folder
☐	sparkwc.py	2.73 KB	binary/octet-stream

For example, here I uploaded sparkwc.py, and my link would be gs://dataproc-e4225d08-23a8-481f-aff9-5205024119a5-us/sparkwc.py

Step 4: submit the job

Similar to our Hadoop example. Use the settings below. Remember to use your own gs:// link, not mine.

Cluster

cluster-1

Job type

PySpark

Main python file ?

gs://dataproc-e4225d08-23a8-481f-aff9-5205024119a5-us/sparkwc.py

Additional python files (Optional)

Enter file path, for example, hdfs://example/example.py

Jar files (Optional) ?

Enter file path, for example, hdfs://example/example.jar

Arguments (Optional) ?

Press <Return> to add more arguments

Properties (Optional) ?

+ Add item

Labels (Optional) ?

Step 5: browse the result

Once the job is done (should be around 2~10 minutes). Go back to Big Query and explore the result.

New Query ?

```
1 select word, word_count from [cuong-project1:wc_class.wc1] order by word_count desc
```

Ctrl + E

RUN QUERY

Save Query

Save View

Format Query

Show Options

Query complete (20.2s elapsed)

Results	Explanation	Job Information	Download as CSV	Download as JSON
Row	word	word_count		
1	the	29801		
2	and	27529		
3	i	21029		
4	to	20957		
5	of	18514		